Process Simulation for Machine Reservation in Cloud Manufacturing

Philipp Waibel*†, Svetoslav Videnov*, Michael Borkowski*, Christoph Hochreiner*, Stefan Schulte* and Jan Mendling†
*Distributed Systems Group, TU Wien, Austria
{p.waibel, s.videnov, m.borkowski, c.hochreiner, s.schulte}@infosys.tuwien.ac.at
†Institute for Information Business, WU Wien, Austria
jan.mendling@wu.ac.at

***Abstract*—Cloud manufacturing supports companies to create cross-organizational elastic process landscapes with flexible and scalable processes. In recent years, the Business Process Model and Notation language gained interest in the cloud manufacturing domain not only as a modeling language but also as a base for process enactment. During this enactment of manufacturing processes, accurate knowledge about when a machine is needed is of great importance. While such planning is simple in a static production environment, it becomes a challenging task in an elastic cloud manufacturing environment with a variety of flexible and scalable processes.**

Hence, we propose an approach that performs enactment simulations before an actual manufacturing process is carried out. The goal is to reserve manufacturing machines at the time at which they will be needed in a machine reservation timetable. By discussing different use case scenarios, we further elaborate on the benefits of our approach.

***Index Terms*—Business Process Enactment, Manufacturing Process Simulation, Cloud Manufacturing, BPMN**

I. Introduction

Today's manufacturing companies are under constant pressure to react to ever changing customer demands with as little delay as possible. This challenge originates from the current trend towards mass customization, fast-changing order cycles or the goal to minimize the time-to-market [1], [2].

To address these trends, a number of novel business and manufacturing concepts have emerged. The concept of *cloud manufacturing* is one of them [3], [4]. Cloud manufacturing transfers well-known principles of cloud computing to the manufacturing domain. The fundamental principle in cloud manufacturing is that companies can create virtualized services of their real-world manufacturing assets, i.e., hard resources (e.g., manufacturing machines) and soft resources (e.g., software) [5], [6]. These virtualized services are then offered as services based on a pay-per-use model to other companies in a marketplace. This sharing of manufacturing assets facilitates the creation of cross-organizational elastic manufacturing process landscapes with manufacturing processes that can react flexibly towards changing process requirements [3].

The possibility to lease a machine and to offer a machine for leasing leads to the requirement of knowing when exactly a machine is needed [7]. Meaning that on the provider side, i.e., the company that offers the machine, it must be known when a machine is not occupied by any manufacturing process so that it can be lent to other companies. On the side of the consumer, i.e., the company that wants to lease a machine, it is required to know when and for how long this particular machine will be needed.

By analyzing a manufacturing process before the actual real-world enactment of the process starts, a reservation of the machines, which are required for the enactment of the process, at the time when they are needed, can be performed. Furthermore, by maintaining a machine reservation timetable, which holds the machine reservation information, it can be determined when a machine is available for leasing and when another process occupies it. Moreover, during the enactment of the manufacturing process, a machine reservation timetable helps to ensure that the machine is available when a process needs it. Thus, the risk of stalemate and high waiting times can be reduced.

The potential of using process models expressed in the Business Process Model and Notation (BPMN) is acknowledged in recent works in the manufacturing domain [8]–[13]. In addition to the process modelling capability, BPMN also offers simulation support [14], which can be exploited for scheduling in a cloud manufacturing scenario. Up until now, a systematic approach to support the reservation of manufacturing machines by BPMN process simulation is missing.

In this paper, we present a method based on pre-enactment simulation for reserving the manufacturing machines in a machine reservation timetable. This timetable specifies already before the actual real-world enactment starts when the machine will be needed. This method was developed as part of the manufacturing process enactment environment of the CREMA framework [4], [15]. Notably, due to the generic nature of the presented approach, it is not bound to the CREMA framework and can also be used independently.

The remainder of the paper is structured as follows: Section II discusses the relevant background information. Subsequently, we discuss a motivational scenario of our work in Section III. Section IV presents the design of our pre-enactment simulation method and the implemented tool support is presented in Section V. In Section VI, we assess the usability of the proposed approach by discussing an industry use case, following with the discussion of the related work in Section VII. Finally, Section VIII concludes this paper.

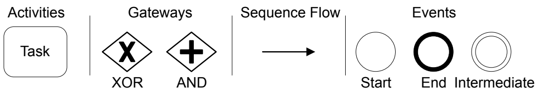


Fig. 1. Subset of the BPMN 2.0 Core Elements.

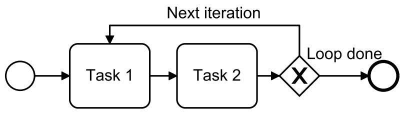


Fig. 2. Example Process with a Loop.

II. Background

BPMN is a high-level and graphical modeling language for business processes. The latest version is BPMN 2.0 [16]. Fig. 1 shows a subset of the BPMN 2.0 core elements and Fig. 2 an example process in BPMN 2.0 notation. In the following, we will discuss the elements shown in Fig. 1. The full specification of BPMN 2.0 can be found in [16].

In BPMN, a unit of work is represented by an *activity*. An activity can be a single task (e.g., a software task or a human-executed task) or contain a sub-process. The activities of a process are combined by *sequence flows* and *gateways* (e.g., AND or XOR) that define the execution order of the activities. Finally, a process can also react to *events*. Beside the start and finish events, intermediate events can be handled during the process execution, e.g., wait for a defined time. As a data format, BPMN is using XML. The BPMN standard further allows the extension of the core elements while still preserving BPMN-compliancy.

The rich set of core elements and the extensibility of BPMN makes it an excellent choice for describing manufacturing processes in the field of cloud manufacturing. In such a cloud manufacturing scenario, a virtualized description of a real-world manufacturing asset or service can be mapped to an activity in a process model as described in [15]. Moreover, this mapping can be done in a manual way, e.g., at design time, or in an automatic way, e.g., at design or run-time. In the latter case, an automatic selection of the best-fitting service to a BPMN activity can be performed as described in [11], [15].

In this way, BPMN allows the precise definition of manufacturing processes and provides a base to enact the designed processes. In addition, due to the high extensibility of BPMN 2.0, it is well suited for our use case.

III. Motivational Scenario

To motivate our work, we use a scenario from the manufacturing industry. This motivational scenario is inspired by the use cases of the EU H2020 project CREMA [15]. The scenario describes a company, called *CompanyOne*, that produces various products, all of which are highly customizable to meet specific requirements. To be able to cope with fluctuating order demands for the products, i.e., a lot of orders in peak times and only a few in off-peak times (e.g., during the holidays), the company is using cloud manufacturing as the foundation for a cross-organizational elastic process landscape. This landscape combines different manufacturing plants and suppliers.

CompanyOne uses BPMN 2.0 as process modeling language. The designed manufacturing process models are used for analyzing and planning purposes and as a base for the enactment of the manufacturing process.

In peak times, CompanyOne leases additional manufacturing capacities from other companies by using the concepts of cloud manufacturing. This way, CompanyOne can rapidly increase their productivity to fulfill their customer's orders without having to buy additional manufacturing machines that are not required in off-peak times. In off-peak times, not all manufacturing machines owned by CompanyOne, are working to capacity. In such times, CompanyOne offers the unused capacities to other companies.

To know when a machine is needed and when it is available, CompanyOne maintains a machine reservation timetable. This machine reservation timetable holds for each manufacturing machine the information when it is needed, i.e., start and end time, and for which process it is needed at this time. This machine reservation timetable is then further used to schedule the enactment of the manufacturing processes, to offer the unused machines to other companies, and to plan other tasks, e.g., machine maintenance. Additionally, the information when a machine is required for the enactment of a process is used to reserve and lease machines from other companies if needed.

Fig. 3 shows a simplified version of the presented scenario. The figure shows the machine reservation timetable for four machines, called M1–M4, of CompanyOne and two machines (M1 and M2) of the company Supplier 1. Furthermore, the picture shows that CompanyOne reserved M1 from Supplier 1 for one hour. Notably, Fig. 3 shows only a small excerpt of CompanyOne's manufacturing network.

As can be seen in the scenario, the knowledge of when a machine is needed is of great importance in cloud manufacturing. Since BPMN 2.0 gained more and more attention in the manufacturing domain [8]–[12], the usage of BPMN 2.0 models as a base for the creation of the machine reservation timetable seems to be an obvious choice.

IV. Approach

The idea of our approach is to perform a pre-enactment simulation before the real-world enactment of a manufacturing process starts. During this pre-enactment simulation, the manufacturing machines used by this particular process, are reserved for the time they will be needed. Thereby, it is guaranteed that the machines are available at those times when needed and not occupied by another process.

In the following, we discuss the general approach of the pre-enactment simulation and the machine reservation.

A. General Approach

The pre-enactment simulation and reservation of the manufacturing machines are performed for each enacted manufacturing process separately. When the enactment of a process is requested, our approach first simulates the process enactment

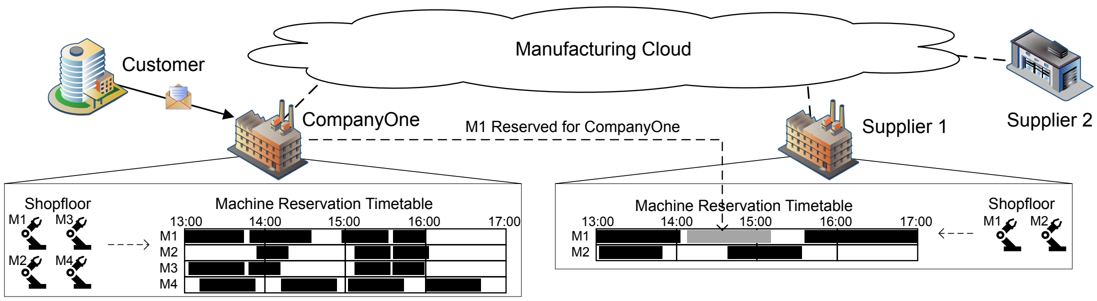


Fig. 3. Motivational Scenario Including the Machine Reservation Timetables of CompanyOne and Supplier 1.

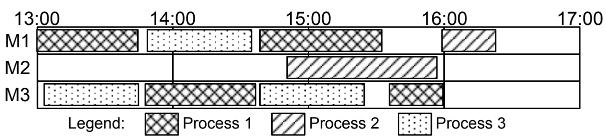


Fig. 4. Machine Reservation Timetable Example.

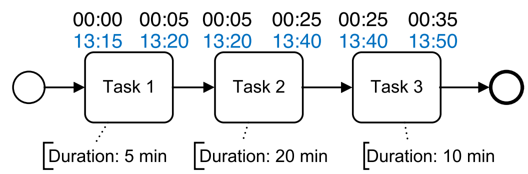


Fig. 5. Reservation Time Calculation Steps.

and uses this simulation to calculate the times, i.e., start and end time, when a manufacturing machine is required. Those times are then used to reserve the machine in a machine reservation timetable for this particular process. After the simulation is finished, the actual real-world enactment starts. An example machine reservation timetable is depicted in Fig. 4 which shows the reservation times for three machines (M1–M3) and three processes. The pre-enactment simulation approach is independent of the machine reservation timetable which can be represented by any conventional timetable. Therefore, in the work at hand, we only discuss the pre-enactment simulation.

If the machine is already reserved for another process, corresponding countermeasures have to be triggered, e.g., rescheduling of the process enactment or searching for an alternative manufacturing machine by using the principles of cloud manufacturing. Possible countermeasures are presented in [11] and [12] and are not further discussed in the following.

For the pre-enactment simulation, our approach extends the default BPMN 2.0 XML schema with two elements, namely *enactmentDuration* and *reservationRequired*. Listing 1 shows this extension with example values.

Listing 1. BMPN 2.0 XML Extension for Machine Reservation.

```
1 <bpmn:serviceTask id="ProcessStep_Name">
2   <bpmn:extensionElement>
3     <enactmentDuration>1200</enactmentDuration>
4     <reservationRequired>true</reservationRequired>
5   </bpmn:extensionElement>
6   ...
7 </bpmn:serviceTask>
```

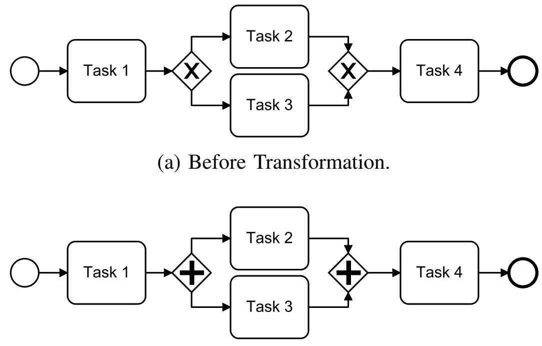
The first extension element, *enactmentDuration* (line 3 in Listing 1), adds the enactment duration of the real-world manufacturing machine in seconds, e.g., 1200 sec. This time defines the duration the machine needs to fulfill its task. This real-world enactment duration has to be known upfront, e.g., via historical sensor data, and defined for each process task.

The second extension element, *reservationRequired* (line 4 in Listing 1), adds the possibility to define if this particular process task should be reserved in the machine reservation timetable (true) or not (false). With this option, the approach considers that in a manufacturing process not all process tasks have to be reserved, e.g., software services that can be invoked several times in parallel or short-running manufacturing steps for which no particular machine is needed. However, also the process tasks that do not require a reservation are considered for the calculation of the reservation times since also those tasks contribute to the total enactment time.

To calculate the correct reservation times for each machine, it has to be considered that the real-world enactment starts after the pre-enactment simulation. However, at the time of the simulation, the correct starting time of the real-world enactment is still unknown. To overcome this, the times calculated during the simulation are delta values starting with 00:00 at the beginning of the process, as shown in Fig. 5 (in black digits). To obtain the real-world enactment times, the current time is added to the calculated times shortly before the real-world process enactment starts. This step is depicted in Fig. 5 with blue digits and an example real-world enactment starting time of 13:15. The resulting times are then used to reserve the machines in the machine reservation timetable.

To consider unforeseeable circumstances during the real-world enactment, e.g., a longer machine enactment duration or a delay due to congestion, the reservation times are compared to the real-world enactment times before the enactment of each process task. If the times differ, the simulation is repeated to update the reservations in the machine reservation timetable.

Eventually, the machine reservation timetable contains the information when a machine will be needed during the real-world enactment of a process. However, if the process is not a sequential workflow, as depicted in Fig. 5, but a more complex



(a) Before Transformation.

(b) After Transformation.

Fig. 6. Static Simulation Transformation.

composition with decision gateways, as depicted in Fig. 6a, the complexity of the pre-enactment simulation increases. Gateways with different outgoing branches pose a certain uncertainty: At the time of the pre-enactment simulation, it is not known which branch the real-world enactment will take when a decision gateway is reached. We propose two strategies for handling such uncertainties, as described in the next two subsections.

B. Static Simulation

The idea of the *static simulation* is to transform each decision gateway into a parallel gateway. By doing this, the approach considers all possible branches during the pre-enactment simulation.

For this, the static simulation loads the manufacturing process model and transforms it, i.e., substitutes all decision gateways by parallel gateways. An example transformation can be seen in Fig. 6, where Fig. 6a shows the original process, i.e., before the transformation, and Fig. 6b the process after the transformation.

Subsequently, the transformed model is used for the pre-enactment simulation and, thus, for the calculation of the manufacturing machine's reservation times. Afterward, the original model is used for the real-world enactment of the manufacturing process.

This approach solves the problem of not knowing which branch will be taken at runtime. However, if the process model contains a loop, as depicted in Fig. 2, the number of loop iterations will still be uncertain. For this, the approach uses a counter defining how often a loop should be simulated. Thereby, the amount of loop iterations has to be known by background knowledge about the process, e.g., by taking into account historical data about former process enactments.

The benefits of this approach are the simplicity and performance. The negative aspect of this static simulation approach is the fact that all manufacturing machines, required by the process, are reserved. This includes machines used in branches which are not enacted at all during the real-world enactment. For instance, the static simulation approach reserves the manufacturing machines used for Task 2 and Task 3 in the process depicted in Fig. 6a. However, since the branch is chosen by an exclusive gateway only one of the tasks is executed, e.g., Task 2. This leads to an unnecessary reservation for the other task, e.g., Task 3.

However, if the branches after the decision gateway contain only process steps for which a reservation is not needed, the before discussed negative aspect is negligible. For instance, in Fig. 6a if Task 2 and 3 do not require a reservation but Task 4 needs one. Then the fact that the static simulation method simulates both branches is negligible.

To prevent the superfluous reservation of unneeded manufacturing machines, in the following, we suggest a more dynamic simulation approach.

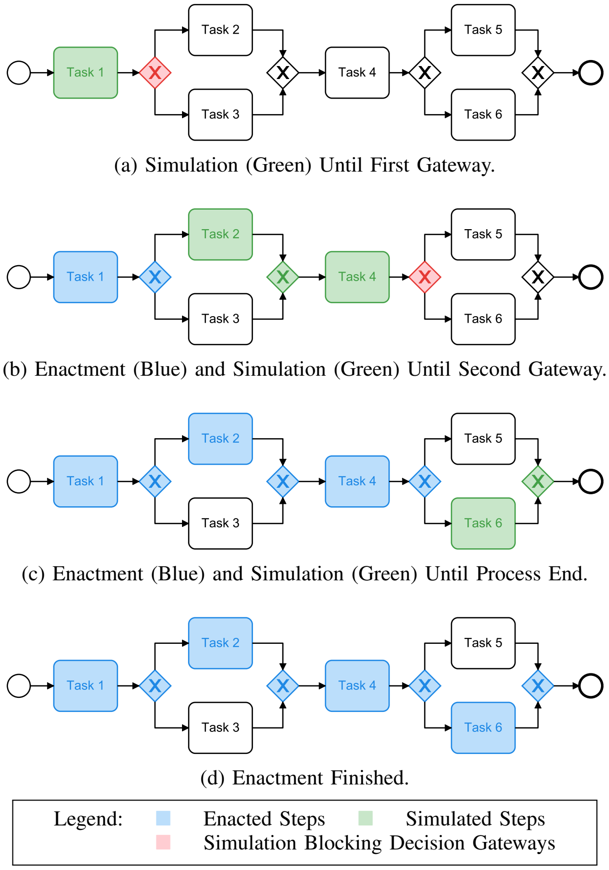
C. Semi-Dynamic Simulation

In comparison to the static simulation approach presented in Section IV-B, the *semi-dynamic simulation* approach uses a combination of the pre-enactment simulation and real-world enactment to overcome the problem of not knowing the path after a decision gateway.

The basic idea of the semi-dynamic simulation approach is to perform the pre-enactment simulation – and therefore the calculation of reservation times – for as long as it is possible. This means that the simulation starts at the beginning of the process and continues until the first decision gateway, for which it is unknown which branch will be taken during the real-world enactment phase, is reached. This is depicted in Fig. 7a, where green symbolizes the tasks that have been simulated and red symbolizes the decision gateway for which the subsequent path is unknown. At this point, the simulation stops and a switch to the real-world enactment takes place. This real-world enactment starts at the beginning of the process and continues until the first decision gateway is reached, as depicted in Fig. 7b, where blue symbolizes already executed tasks.

When the enactment reaches the first decision gateway, and a branch is selected for the enactment, based on the process state, the simulation can continue. Hence, the real-world enactment pauses, and a switch back to simulation takes place, which continues until another decision gateway or the end of the process is encountered. If another decision gateway is reached (as depicted in Fig. 7b), the enactment simulation pauses again, and the real-world enactment continues. The real-world enactment starts now at the point where it stopped, i.e., after the last decision gateway, and continues until the new decision gateway (as depicted in Fig. 7c) or until the end of the process is reached. This is repeated until the end of the process is reached.

With this stepwise pre-enactment simulation, the approach overcomes the problem of the superfluous manufacturing machine reservations, as it happens with the static simulation approach discussed in Section IV-B. Furthermore, the semi-dynamic simulation approach also solves the problem of the unknown amount of loop iterations also discussed in Section IV-B, since each iteration of a loop in the process is handled separately.



(a) Simulation (Green) Until First Gateway.

(b) Enactment (Blue) and Simulation (Green) Until Second Gateway.

(c) Enactment (Blue) and Simulation (Green) Until Process End.

(d) Enactment Finished.

Fig. 7. Semi-Dynamic Simulation Steps.

It has to be noted that, due to the continuous switching between the pre-enactment simulation and real-world enactment, the overall performance is lower than for the static simulation. Furthermore, in comparison to the static method, depending on the process, not all manufacturing machines might be reserved immediately at the beginning of the enactment but after some steps are already performed. This may lead to the fact that some machines, which could have been the best choice prior to process enactment, are not available any longer when the semi-dynamic simulation is performed.

V. Tool Support

In general, the simulation approaches presented in Section IV could be integrated into any process planning software. As a proof-of-concept, we realize the approaches as a prototype on top of the Camunda BPMN Workflow Engine[1], which is an open source business process management system.

Camunda provides the possibility to enact business processes based on BPMN 2.0. Therefore, Camunda provides a good base for our implementation since the envisioned pre-enactment simulation is basically an enactment of the process without the real-world services, but instead, a stepwise calculation of the usage times. Consequently, we extended Camunda in a way that it performs the pre-enactment simulation before the real-world enactment. This implementation is part of the CREMA framework [15]. As machine reservation timetable and automatic countermeasures if a machine is already reserved by another process, we use available components in the CREMA framework.

Our implementation uses and extends the following functionalities of Camunda: (i) the BPMN parser, (ii) the possibility to add a listener to process elements (e.g., decision gateways), (iii) and the enactment of process tasks by external workers. The implemented extensions are discussed in the following:

BPMN Parser: We extend the BPMN parser of Camunda in a way that it adds to each decision gateway a listener called *Decision Gateway Listener*. This listener is called each time a decision gateway is reached during the simulation and real-world enactment.

Decision Gateway Listener: The activity of this listener depends on the current phase of the enactment, i.e., pre-enactment simulation (static or semi-dynamic simulation) or real-world enactment:

- If the pre-enactment simulation of the current enactment is a static simulation, then the behavior of the gateway is changed to that of a parallel gateway behavior as discussed in Section IV-B.
- If the pre-enactment simulation is semi-dynamic:
  - If the current enactment is pre-enactment simulation, a switch to the real-world enactment takes place.
  - If the current enactment is real-world enactment, a switch to the pre-enactment simulation takes place.

External Tasks: The external task functionality of Camunda collects process model tasks that have to be enacted in a queue, according to the order defined in the process model. These tasks are then taken from the queue and enacted by so-called workers. This enactment can be sequential, i.e., one task after another, or in parallel. We implement two different workers:

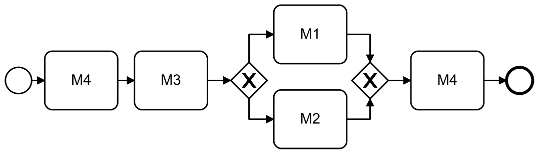
- Pre-Enactment Simulation Worker: This worker is used for the pre-enactment simulation. It performs the calculation of the starting and end times for the reservations as discussed in Section IV-A.
- Real-World Enactment Worker: This worker is responsible for the enactment of the tasks when the real-world enactment is in progress. This means that it is responsible for the actual enactment of the real-world functionality defined in a process task.

In addition, we hold for each process enactment the information in which phase, i.e., simulation or real-world enactment, the enactment currently is. This information is then used in the Decision Gateway Listener and the selection of the external task worker.

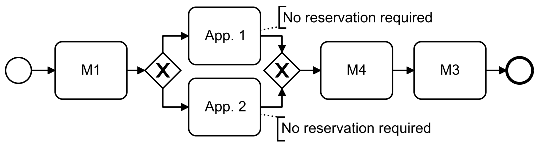
VI. Discussion

To discuss the benefits of the manufacturing machine reservation, enabled by our approach, we use in the following the motivational scenario discussed above. CompanyOne (as introduced in Section III) produces three different products,

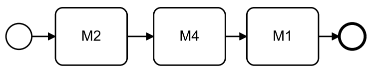
[1]https://camunda.org



(a) Process 1.



(b) Process 2.



(c) Process 3.

Fig. 8. CompanyOne's Manufacturing Processes.

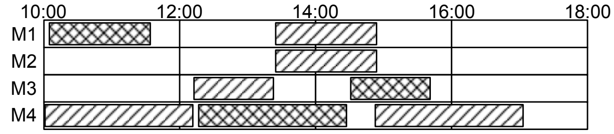
each defined by a process (i.e., *Process 1*, *Process 2*, and *Process 3*) as depicted in Fig. 8. The task names, e.g., M1, defines which machine is used by this task. Additional to the manufacturing machines, Process 2 includes two applications, i.e., App. 1 and App. 2, that do not require a reservation.

In the following, we first discuss the results of enacting all three processes with the static simulation approach in Section VI-A and then the results with the semi-dynamic simulation approach in Section VI-B.

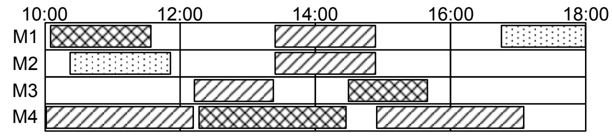
A. Static Simulation

At the beginning of this scenario, the machine reservation timetable is empty since no manufacturing process is running and thus, all machines are free. At the time 10:00 an enactment of Process 1 is requested and shortly after that an enactment of Process 2 is requested. Each request leads to a pre-enactment simulation that reserves the required machines for Process 1 and 2. Fig. 9a shows the resulting machine reservation timetable. As shown in Fig. 8, Process 1 and Process 2 contain one decision gateway each. Since the current pre-enactment simulation mode is applying static simulation, all possible branches are simulated. For Process 1, this means that all machines are reserved, as can be seen in Fig. 9a. For Process 2, this is not the case since the parallel branches contain process steps that do not require a reservation. Only the machines after the join, i.e., M4 and M3, require a reservation as shown in Fig. 9a.

Following this, an enactment request for Process 3 takes place at 10:30. Since machine M4 is already reserved for the enactment of Processes 1 and 2, an alternative machine is required. In our scenario, company Supplier 1 is able to provide an alternative machine (M6), which is then reserved for Process 3. The remaining machines for Process 3 (M2 and M1) are not reserved for other processes. Hence, they are reserved at CompanyOne. The resulting machine reservation timetable is depicted in Fig. 9b, for CompanyOne, and Fig. 10, for Supplier 1. However, since Supplier 1 is an external



(a) Machine Reservation Timetable after the Enactment of Process 1 and 2 was Requested.



(b) Machine Reservation Timetable after the Enactment of Process 3 was Requested.

Fig. 9. Machine Reservation Timetables with Static Simulation.

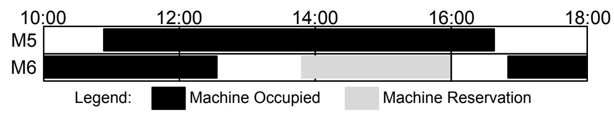


Fig. 10. Machine Reservation Timetable of Supplier 1 after CompanyOne Reserved a Machine.

company, CompanyOne only sees when a machine is occupied (as depicted in Fig. 10 by a black rectangle) or free and not for which process the machine is used, due to privacy concerns.

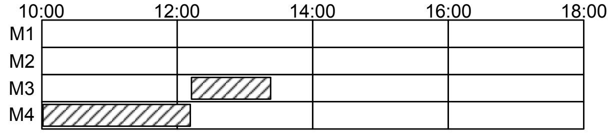
This example shows that the static simulation approach can help to identify situations where the enactment of a process would stagnate due to a lack of available machines. However, as explained in Section IV-B, the downside of this approach is that all machines are reserved, including the machines of the branch that is not enacted, i.e., M1 and M2 in Process 1 (Fig. 8a).

Furthermore, this case study shows that a machine reservation timetable and the presented simulation approaches provide useful information for companies operating in a cloud manufacturing network. First, it helps the companies to plan and enact their manufacturing processes without the risk of a bottleneck where two processes need the same machine at the same time. Second, the machine reservation timetable helps the companies to analyze their machine usage and maximize their machine utilization by lending unused machines to other companies. This is achieved without having to risk stagnation of already running processes.

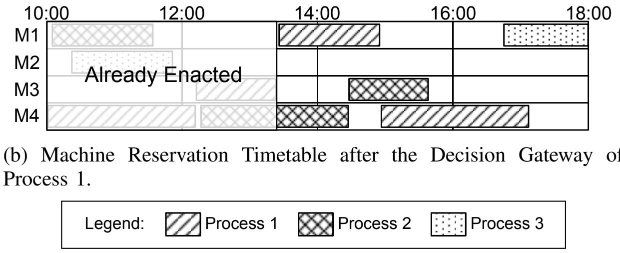
To determine the performance of the presented simulation approach, we enact Process 1, 2, and 3 six times each, by using the CREMA framework and the static simulation. During this enactment, we measure the duration of the simulation. The resulting average duration of Process 1 and 2 is $18.17s$ ($\sigma = 1.32s$), and of Process 3 is $13.33s$ ($\sigma = 1.53s$).

B. Semi-Dynamic Simulation

By switching the pre-enactment simulation mode to semi-dynamic simulation, the machine reservation of Process 1



(a) Machine Reservation Timetable after the Initial Pre-Enactment Simulation of Process 1.



(b) Machine Reservation Timetable after the Decision Gateway of Process 1.

Fig. 11. Machine Reservation Timetables with Semi-Dynamic Simulation.

is performed in two phases. The first phase simulates until the decision gateway and the second phase simulates the remaining steps. After the initial pre-enactment simulation of Process 1, the machine reservation timetable holds only the reservation of M4 and M3 as depicted in Fig. 11a. After the simulation, the real-world enactment starts and continues until the decision gateway is reached.

As soon as the decision gateway of Process 1 is reached by the real-world enactment, a switch to the pre-enactment simulation takes place. Then, the remaining steps of the process are simulated since it is now known which branch the real-world enactment will take. In the example presented in Fig. 11b, the upper branch of Process 1 (Fig. 8a) is selected, i.e., M1, resulting in the timetable shown in Fig. 11b.

The difference of the semi-dynamic simulation in comparison to the static simulation can also be seen by observing the enactment of Process 2 (Fig. 8b). Since Process 2 also contains a decision gateway, the semi-dynamic simulation stops at the gateway and waits for the real-world enactment to reach the gateway. However, since both branches of Process 2 contain tasks without the requirement of a reservation, this is not necessary, and the simulation could continue as performed by the static simulation. Thus, the static simulation is better suited for Process 2, since it reserves machines M4 and M3 already at the start of the enactment, as shown in Fig. 9a.

Also for the semi-dynamic simulation, we determine the simulation duration by enacting all three processes six times each. The resulting average duration of Process 1 and 2 is $21.33s$ ($\sigma = 2.06s$), and for Process 3 is $13.67s$ ($\sigma = 0.58s$).

This discussion shows that the semi-dynamic simulation can help to prevent the reservation of unnecessary machines, in comparison to the static simulation. However, it also shows that in some cases the static simulation, which offers a better performance regarding the simulation duration, is better suited.

VII. Related Work

The research presented in this paper relates to three major streams of work: applications of BPMN in manufacturing, process scheduling and process simulation.

First, the usage of BPMN as a modeling language for manufacturing processes has increased in the recent years. For instance, [9] and [10] discuss how BPMN can be applied to the manufacturing domain by proposing extensions to the BPMN core elements. The authors of [8] compare BPMN 2.0 with other manufacturing process notations and Prades et al. [17] propose a methodology to model processes in BPMN with focus on the cooperation between the enterprise management and the manufacturing operations level.

In [12], Mazzola et al. present a cloud-based execution environment for processes modeled in BPMN for the manufacturing domain. They also take into account exceptions during the enactment of the business processes and perform just-in-time compensations for these exceptions. Moreover, in [13], Skarlat et al. propose a methodology and instrumentation toolset for BPMN in cloud manufacturing.

Second, the scheduling of business processes is defined as the problem to find an execution sequence for the process so that its tasks finish under given constraints, e.g., timing, causality, and resource constraints [18], [19]. Theoretical foundations of process scheduling to the fields of Operations Management and Operations Research are discussed in [20]. Additional surveys in these fields can be found in [21], [22]. In these areas, jobs have to be efficiently scheduled on a number of machines [23], [24]. Process scheduling is also discussed in the area of answer set programming [25], [26]. Manufacturing machine scheduling in the field of cloud manufacturing can be found in [6], [27], [28]. The authors describe different approaches for production task scheduling on different manufacturing machines. In comparison to these works, we make use of BPMN as a modeling language. This can help to increase the understandability of the processes not only on manufacturing operations level but also on enterprise management level [9], [17]. Also, we obtain a formal basis for simulation via the behavioral semantics of gateways.

Third, there are several approaches to process simulation [14]. For instance, prominent usage of simulation includes determining the impact of business processes on the performance of information systems [29] or using simulation to analyze and improve the operational performance of business processes [30]. In the manufacturing domain, the simulation of processes is used primarily to assess the performance of the processes and to evaluate alternatives [31]–[34]. In comparison to our work, the simulation of business processes is not used as a tool during operation to create a machine reservation timetable, but rather as a separate analysis tool at design time. In our case, the simulation and the reservation of the machines help us to guarantee that, for each process enactment, the required machines are available and, thereby, guarantee the consistent enactment of the process. This can not be guaranteed by a simulation at design time since the environment and circumstances could change between the design time and run-time.

All of these approaches target either the usage of BPMN for the manufacturing domain, the scheduling of processes, or the simulation of process enactments for analyzing purposes. To the best of our knowledge, no approach in the present literature uses all of these techniques to create a machine reservation timetable for business processes designed with BPMN 2.0.

VIII. Conclusion

In this paper, we have presented an approach for cloud manufacturing that performs a pre-enactment simulation before the real-world enactment starts. This pre-enactment simulation calculates the times during which a machine will be used by the real-world enactment. Eventually, those times are used to reserve the manufacturing machine in a machine reservation timetable for this particular process. Our discussion highlights that our machine reservation approach supports companies to maximize the utilization of their machines by offering them to other companies when they are not needed.

In our future research, we will focus on how our solutions to the problem of the path selection after a decision gateway can be improved. To this end, we investigate the use of prediction techniques and the usage of historical data. Furthermore, we aim to extend the set of BPMN elements that our approach supports and focus on the privacy aspects of our approach.

Acknowledgments

This work is partially funded by COMET K1, FFG – Austrian Research Promotion Agency, within the Austrian Center for Digital Production.

References

[1] D. Zuehlke, "Smartfactory – towards a factory-of-things," *Annual Reviews in Control*, vol. 34, no. 1, pp. 129–138, 2010.

[2] D. Matt and E. Rauch, "Design of a network of scalable modular manufacturing systems to support geographically distributed production of mass customized goods," *Conf. on Intelligent Computation in Manufacturing Engineering*, vol. 12, pp. 438–443, 2013.

[3] X. Xu, "From cloud computing to cloud manufacturing," *Robotics and computer-integrated manufacturing*, vol. 28, no. 1, pp. 75–86, 2012.

[4] S. Schulte, P. Hoenisch, C. Hochreiner, S. Dustdar, M. Klusch, and D. Schuller, "Towards process support for cloud manufacturing," in *Int. Enterprise Distributed Object Computing Conf.* IEEE, 2014, pp. 142–149.

[5] Y. Lu, X. Xu, and J. Xu, "Development of a hybrid manufacturing cloud," *J. of Manufacturing Sys.*, vol. 33, no. 4, pp. 551–566, 2014.

[6] Y. Lu and X. Xu, "Process and production planning in a cloud manufacturing environment," in *Int. Manufacturing Science and Engineering Conf.* American Society of Mechanical Engineers, 2015.

[7] J. Lartigau, L. Nie, X. Xu, D. Zhan, and T. Mou, "Scheduling methodology for production services in cloud manufacturing," in *Int. Joint Conf. on Service Sciences*. IEEE, 2012, pp. 34–39.

[8] A. García-Domínguez, M. Marcos, and I. Medina, "A comparison of BPMN 2.0 with other notations for manufacturing processes," in *AIP Conf. Proceedings*, vol. 1431, 2012, pp. 593–600.

[9] S. Zor, F. Leymann, and D. Schumm, "A proposal of bpmn extensions for the manufacturing domain," in *Int. Conf. on Manufacturing Sys.*, 2011.

[10] S. Zor, K. Görlach, and F. Leymann, "Using bpmn for modeling manufacturing processes," in *Int. Conf. on Manufacturing Sys.*, 2010, pp. 515–522.

[11] L. Mazzola, P. Kapahnke, and M. Klusch, "ODERU: Optimisation of semantic service-based processes in manufacturing," in *Knowledge Engineering and Semantic Web*. Springer, 2017, pp. 337–346.

[12] L. Mazzola, P. Kapahnke, P. Waibel, C. Hochreiner, and M. Klusch, "FCE4BPMN: On-demand qos-based optimised process model execution in the cloud," in *Int. Conf. on Engineering, Technology and Innovation/ Int. Technology Management Conf.*, 2017, pp. 319–328.

[13] O. Skarlat, M. Borkowski, and S. Schulte, "Towards a methodology and instrumentation toolset for cloud manufacturing," in *Int. Workshop on Cyber-Physical Production Sys.* IEEE, 2016, pp. 1–4.

[14] M. Dumas, M. L. Rosa, J. Mendling, and H. A. Reijers, *Fundamentals of Business Process Management, Second Edition*. Springer, 2018.

[15] S. Schulte, M. Borkowski, C. Hochreiner, M. Klusch, A. Murguzur, O. Skarlat, and P. Waibel, "Bringing cloud-based rapid elastic manufacturing to reality with crema," in *Workshop on Intelligent Sys. Configuration Services for Flexible Dynamic Global Production Networks at the Int. Conf. on Interoperability for Enterprise Sys. and Applications*, 2016, pp. 407–413.

[16] OMG, *Business Process Model and Notation (BPMN), Version 2.0*, Object Management Group Std., Rev. 2.0, January 2011, (Accessed 24-March-2018). [Online]. Available: http://www.omg.org/spec/BPMN/2.0

[17] L. Prades, F. Romero, A. Estruch, A. Garcia-Dominguez, and J. Serrano, "Defining a methodology to design and implement business process models in BPMN according to the standard ANSI/ISA-95 in a manufacturing enterprise," *Procedia Engineering*, vol. 63, pp. 115–122, 2013.

[18] J. Eder, E. Panagos, H. Pozewaunig, and M. Rabinovich, "Time management in workflow systems," in *Int. Conf. on Business Information Systems*. Springer, 1999, pp. 265–280.

[19] A. Avanes and J.-C. Freytag, "Adaptive workflow scheduling under resource allocation constraints and network dynamics," *VLDB Endowment*, vol. 1, no. 2, pp. 1631–1637, 2008.

[20] S. Schulte, C. Janiesch, S. Venugopal, I. Weber, and P. Hoenisch, "Elastic business process management: State of the art and open challenges for bpm in the cloud," *Future Generation Computer Sys.*, vol. 46, pp. 36–50, 2015.

[21] J. Xu and D. L. Parnas, "Scheduling processes with release times, deadlines, precedence, and exclusion relations," *IEEE Transactions on Soft. Engineering*, vol. 16, no. 3, pp. 360–369, 1990.

[22] E. Mokotoff, "Parallel machine scheduling problems: A survey," *Asia - Pacific J. of Operational Research*, vol. 18, no. 2, pp. 193–242, 2001.

[23] K. R. Baker and D. Trietsch, *Principles of sequencing and scheduling*. John Wiley & Sons, 2013.

[24] G. Baggio, J. Wainer, and C. Ellis, "Applying scheduling techniques to minimize the number of late jobs in workflow systems," in *Symposium on Applied computing*. ACM, 2004, pp. 1396–1403.

[25] M. L. Rosa, P. Loos, and O. Pastor, Eds., *Business Process Management Forum - BPM Forum 2016, Rio de Janeiro, Brazil, September 18-22, 2016, Proceedings*, ser. Lecture Notes in Business Information Processing, vol. 260. Springer, 2016.

[26] G. Havur, C. Cabanillas, J. Mendling, and A. Polleres, "Resource allocation with dependencies in business process management systems," in *Business Process Management Forum - BPM Forum 2016, Rio de Janeiro, Brazil, September 18-22, 2016, Proceedings*, 2016, pp. 3–19.

[27] W. Li, C. Zhu, L. T. Yang, L. Shu, E. C.-H. Ngai, and Y. Ma, "Subtask scheduling for distributed robots in cloud manufacturing," *IEEE Sys. J.*, vol. 11, no. 2, pp. 941–950, 2017.

[28] Y. Liu, X. Xu, L. Zhang, L. Wang, and R. Y. Zhong, "Workload-based multi-task scheduling in cloud manufacturing," *Robotics and Computer-Integrated Manufacturing*, vol. 45, pp. 3–20, 2017.

[29] R. Heinrich, P. Merkle, J. Henss, and B. Paech, "Integrating business process simulation and information system simulation for performance prediction," *Soft. & Sys. Modeling*, vol. 16, no. 1, pp. 257–277, 2017.

[30] S. Bisogno, A. Calabrese, M. Gastaldi, and N. Levialdi Ghiron, "Combining modelling and simulation approaches: How to measure performance of business processes," *Business Process Management J.*, vol. 22, no. 1, pp. 56–74, 2016.

[31] K. D. Barber, F. W. Dewhurst, R. Burns, and J. Rogers, "Business-process modelling and simulation for manufacturing management: A practical way forward," *Business Process Management J.*, vol. 9, no. 4, pp. 527–542, 2003.

[32] K. Hauser and D. Paper, "Simulation of business re-engineering processes: Case study of a united states motor manufacturing company," *Int. J. of Management*, vol. 24, no. 4, pp. 676–687, 2007.

[33] M. Jahangirian, T. Eldabi, A. Naseer, L. K. Stergioulas, and T. Young, "Simulation in manufacturing and business: A review," *European J. of Operational Research*, vol. 203, no. 1, pp. 1–13, 2010.

[34] P. Bocciarelli, A. D'Ambrogio, A. Giglio, E. Paglia, and T. Panetti, "A cloud-based service-oriented architecture for business process modeling and simulation," in *CEUR Workshop Proceedings*, 2017, pp. 87–93.